

*Article*

# Gaussian Multiscale Aggregation applied to Segmentation in Hand Biometrics

Alberto de Santos Sierra \*, Carmen Sánchez Ávila, Javier Guerra Casanova and Gonzalo Bailador del Pozo

Group of Biometrics, Biosignals and Security, Universidad Politécnica de Madrid, Campus de Montegancedo s/n, 28223 Pozuelo de Alarcón, Madrid, Spain

\* Author to whom correspondence should be addressed; alberto@cedint.upm.es, Tel. +34 91 452 49 00, ext 1723, Fax: + 34 91 336 45 01

*Version November 6, 2011 submitted to Entropy. Typeset by L<sup>A</sup>T<sub>E</sub>X using class file mdpi.cls*

---

**Abstract:** This paper presents an image segmentation algorithm based on gaussian multiscale aggregation oriented to hand biometric applications. The method is able to isolate the hand from a wide variety of background textures such as carpets, fabric, glass, grass, soil or stones. The evaluation was carried out by using a publicly available synthetic database with 408000 hand images in different backgrounds, comparing the performance in terms of accuracy and computational cost to two competitive segmentation methods existing in literature, namely Lossy Data Compression (LDC) and Normalized Cuts (NCuts). The results highlight that the proposed method outperforms current competitive segmentation methods with regard to computational cost, time performance, accuracy and memory usage.

**Keywords:** Hand Biometrics, Multiscale Aggregation, Image Segmentation, Image Processing, Biometrics, Security.

---

## 1. Introduction

Hand biometrics is receiving an increasing attention at present because of their huge applicability in daily scenarios and the relation between user acceptance and identification/verification rates [1,2].

The characteristics of this biometric technique in terms of non-invasiveness and acceptability highlight the fact that hand biometrics could be a proper and adequate biometric method for verification and

identification in devices like PC or mobile phones, since hand biometrics system requirements are easily met with a standard camera and hardware processor.

However, as applications requiring hand biometrics tends to contact-less, platform-free scenarios (e.g., smartphones [3]), hand acquisition (capturing and segmentation) is being increased in difficulty. In other words, hand biometrics is evolving from constrained and contact-based scenarios [15,25] to opposite approaches where less collaboration is required from individuals [3,19], providing non-invasive characteristics to this biometric technique, and thus, improving its acceptability.

Consequently, image pre-processing becomes compulsory to tackle with this problem, by providing an accurate segmentation algorithm to isolate hand from background, whatever its nature, and independent from environment and illumination conditions.

Thus, a segmentation method is proposed able to isolate hand from different background, regardless the environmental and illumination conditions.

The proposed approach is based on multiscale aggregation, gathering pixels along scales according to a given similarity gaussian function. This method produces an iterative clustering aggregation, providing a solution for hand image segmentation with a quasi-linear computational cost and an adequate accuracy for biometric applications.

The method has been tested with a synthetic image database, with around 408000 images considering different backgrounds (e.g., soil, skins/fur, carpets, walls or grass) and illumination environments, and compared to two competitive approaches in literature in terms of image segmentation. These approaches are named Lossy Data Compression (LDC) [4] and Normalized Cuts (NCut) [5].

Finally, the layout of the paper remains as follows: Section 2 provides an overview on the current literature, describing the proposed method under Section 3. The database involved in evaluation is presented in Section 4, together with the results, presented in Section 5, providing conclusions and future work in Section 6.

## 2. Literature Review

Segmentation is an important research field in image processing [6], essential in biometric techniques involving image-based data acquisition like hand geometry [7], palmprint [8], hand vein [9], face [10], iris [11], ear [12], gait [13] or handwriting [14].

In fact, the overall performance in terms of identification accuracy relies strongly on the result provided by the segmentation and pre-processing procedure.

Concerning hand-based biometrics, segmentation has received little attention in early works, provided that initial approaches carry out the acquisition procedure in a constrained and homogeneous background [15,16]. This background was selected so that hand segmentation is a trivial task by simple thresholding.

However, as hand biometrics is evolving from contact and peg-based approaches to completely contact-less, peg-free and platform independent scenarios, hand segmentation is increasing its difficulty and complication [17–19].

Several approaches in literature tackle with this problem by providing non-contact, platform-free scenarios but with constrained background, usually employing a monochromatic color, easily distinctive from hand texture by means of simple image thresholding [20–22]. More realistic environments pro-

pose a color-based segmentation, detecting hand-like pixels either based on probabilistic [23], clustering methods [24] or edge detection [15,18,25].

A possible solution for unconstrained and non-homogeneous backgrounds is a segmentation method based on multiscale aggregation [26–30], inspired on the well-known Normalized Cuts approach [5].

The most common applications of this approach consider image segmentation and boundary detection based on texture [29,31], providing accurate results when compared to human segmentation and other competitive approaches in literature [32].

The results obtained by multiscale aggregation in the fields of unsupervised image segmentation are certainly promising [32], and the application of this method for hand segmentation has been recently proposed [3].

Nonetheless, several aspects must be improved in terms of computational cost and memory usage efficiency [3,30,32]. In fact, these methods are strongly dependent on the number of pixels in an image, and only small images are supported. This limitation was partially solved [3,30], providing a quasi-linear segmentation method, described in detail in the following section.

### 3. Gaussian Multiscale Aggregation

The proposed approach attempts to provide an accurate segmentation of a colour hand image. The algorithm strategy consists of aggregating similar nodes according to a specific criteria along different scales until a given goal is met, ensuring that aggregated nodes within segments verify certain properties.

First step of the algorithm consists of providing a particular structure to the amount of elements within the image. Likewise to other methods [30], the proposed algorithms assumes that a given image  $I$  can be represented by a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  where nodes in  $\mathcal{V}$  represent pixels in the image and edges in  $\mathcal{E}$  stands for the structure provided to the set of nodes.

In this approach, the structure on the first scale is assumed to be a 4-neighbourhood strategy, while for subsequent scales, structure is provided by means of Delaunay triangulation [33].

In addition, each node is represented by a similarity function denoted by  $\phi_{v_i}^{[s]}$ , where  $v_i \in \mathcal{V}$  designates a node in graph  $\mathcal{G}$  and  $s$  indicates the scale the element  $v_i$  belongs to. This similarity function is described in terms of relative measures with respect to intensity average and standard deviation.

More in detail,  $\phi_{v_i}^{[s]}$  is represented by a gaussian distribution  $\mathcal{N}(\mu, \sigma)$  where  $\mu$  and  $\sigma$  specify the average and standard deviation neighbour intensity, provided the 4-neighbourhood structure.

Thus, similarity functions leads to the concept of likelihood between nodes in connecting edges, providing a definition of weights within graph  $\mathcal{G}$ .

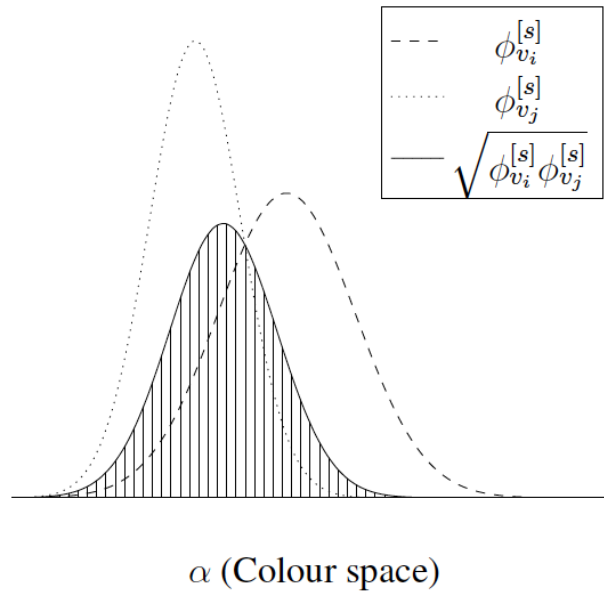
Given a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , the similarity among pair of nodes is provided by means of weights  $\mathcal{W}$ , which are defined for each scale  $s$  as:

$$\mathcal{W}_{i,j}^{[s]} = \int_{\alpha} \sqrt{\phi_{v_i}^{[s]} \phi_{v_j}^{[s]}} d\alpha \quad (1)$$

where  $v_i, v_j \in \mathcal{V}$ ,  $\forall i, j$  and  $\phi_{v_i}^{[s]}, \phi_{v_j}^{[s]}$  represent the similarity function for nodes  $v_i$  and  $v_j$ , respectively. In addition,  $\alpha$  stands for the selected colour space, which in this paper corresponds to the  $a$  layer of the CIELAB (CIE 1976 L\*,a\*,b\*) colour space, due to its ability to describe all visible colors by the human eye [6].

Figure 1 represents two functions  $\phi^{[s]}$  associated to a pair of nodes  $v_i$  and  $v_j$ , showing the weight associated to their similarity (striped region). The higher the similarity between both nodes, the bigger the striped region.

**Figure 1.** Visual representation of two functions  $\phi^{[s]}$  and the weighted  $\mathcal{W}_{i,j}^{[s]}$  associated (striped region).



Therefore, graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$  contains not only structural information on a given scale  $s$  but also relational details about the similarity of each node neighbourhood.

Furthermore,  $\mathcal{W}_{i,j}$  can be regarded as the weight associated to edge  $e_{i,j}$ , so that  $\mathcal{W}_{i,j} = \mathcal{W}(e_{i,j})$ . Notice that weights are not defined for each pair of nodes in  $\mathcal{V}$ , but only for those pairs of nodes with correspondence in edge set  $\mathcal{E}$ .

Some properties can be extracted from the definition of  $\mathcal{W}_{i,j} \in \mathcal{W}$  as the similarity between two nodes  $v_i$  and  $v_j$ , then  $\mathcal{W}_{i,j}$  satisfies  $\forall i, j$ :

1.  $\mathcal{W}_{i,j} \geq 0$
2.  $\mathcal{W}_{i,j} = \mathcal{W}_{j,i}$
3.  $\mathcal{W}_{i,j} = 1 \leftrightarrow \phi_i = \phi_j$

Property (1) results from the definition given by Equation 1, since the integration of two non-negative functions provides a non-negative result. Similarly, property (2) is derived from the commutative product of a function product. Property (3) indicates that maximum value of weight is obtained if and only if nodes  $v_i$  and  $v_j$  have the same similarity distribution.

These former properties stand for each scale  $s$ , although for the sake of simplicity this index was not included on previous notation.

Furthermore, each node  $v_i \in \mathcal{V}$  contains also information on the location within the image in terms of positions, which will be useful in posterior scale aggregation steps.

On the other hand, the essence of this algorithm relies on aggregation, which consists of grouping and clustering those similar nodes/segments in subgraphs, according to some criteria along scales.

The proposed method bases the aggregation procedure on the weights in  $\mathcal{W}$ , given the fact that, those pairs of nodes/subgraphs with higher weights are more similar than those with lower weights, and therefore, those former pairs deserve to be aggregated under a same segment/subgraph. Thus, a function must be defined to provide some order in set  $\mathcal{W}$ , so that posterior subgraphs in subsequent scales contain nodes with high weights and, therefore, high similarity.

Let  $\Omega$  be an ordering function, which orders edges in  $\mathcal{E}$  according to  $\mathcal{W}$ , as follows:

$$\begin{aligned}\Omega : \mathcal{E} &\mapsto \mathbb{R} \\ e &\mapsto \Omega(e)\end{aligned}\tag{2}$$

so that if  $\mathcal{W}_{i,j} = \mathcal{W}(e_{i,j}) \geq \mathcal{W}_{i,k} = \mathcal{W}(e_{i,k})$ , then  $\Omega(e_{i,j}) \geq \Omega(e_{i,k})$ .

In other words, let  $e = \{e_1, \dots, e_m\}$  be a set of edges. If  $\Omega$  is applied to previous set  $e$ , then it is satisfied that  $\Omega(e)_i \geq \Omega(e)_j$ , with  $i \leq j$ ,  $\forall i, j$ , being  $\Omega(e)_i$  the  $i$ th element in the ordered set  $\Omega(e)$ . Concretely,  $\Omega_{\mathcal{W}}$  represent the weight set  $\mathcal{W}$  after  $\Omega$  is applied.

Once the concept of ordering function is introduced, the algorithm aggregates pair of nodes based on this former weight ordering, ensuring that the dispersion of each segment remains bounded. This aggregation criteria is represented by the Equation 3:

$$\delta_{i,j} = \begin{cases} 1, & \sigma_{i,j} \leq \sqrt{\sigma_i \sigma_j} \\ 0, & \text{otherwise} \end{cases}\tag{3}$$

where  $\sigma_{i,j}$  represent the dispersion of aggregating nodes  $v_i$  and  $v_j$ . Despite of selecting the geometric mean as the comparison criteria in previous equation, other methods are possible such as arithmetic mean, generalized mean or harmonic mean. The selection of geometric mean was carried out based on experimental results.

Once pairs of nodes have been ordered and an aggregation criteria have been stated, the Gaussian Multiscale Algorithm aggregates pair of nodes with previous criteria (Equation 4), considering the fact that  $\mathcal{G}_{v_i}^{[s]}$  represents the  $n$ -th graph in scale  $s$ , so that  $v_i \in \mathcal{G}_n^{[s]}$ .

$$\begin{aligned}(\mathcal{G}_{v_i}^{[s]}, \mathcal{G}_{v_j}^{[s]}) &= \\ \begin{cases} (\mathcal{G}_p^{[s]} \delta_{\sigma_{i,j}} + \mathcal{G}_{p+1}^{[s]} \bar{\delta}_{\sigma_{i,j}}, \mathcal{G}_p^{[s]}) & \nexists n/v_i \in \mathcal{G}_n^{[s]}, \nexists m/v_j \in \mathcal{G}_m^{[s]} \\ (\mathcal{G}_n^{[s]}, \mathcal{G}_n^{[s]} \delta_{\sigma_{i,j}} + \mathcal{G}_p^{[s]} \bar{\delta}_{\sigma_{i,j}}) & v_i \in \mathcal{G}_n^{[s]}, \nexists m/v_j \in \mathcal{G}_m^{[s]} \\ (\mathcal{G}_m^{[s]} \delta_{\sigma_{i,j}} + \mathcal{G}_p^{[s]} \bar{\delta}_{\sigma_{i,j}}, \mathcal{G}_m^{[s]}) & v_j \in \mathcal{G}_m^{[s]}, \nexists n/v_i \in \mathcal{G}_n^{[s]} \\ (\mathcal{G}_n^{[s]}, \mathcal{G}_m^{[s]}) & v_i \in \mathcal{G}_n^{[s]}, v_j \in \mathcal{G}_m^{[s]} \end{cases}\end{aligned}\tag{4}$$

In other words, GMA approach aggregates a pair of nodes in scale  $s$  under the same existing segment when at least one of both is already assigned to a segment. In case none has previously assigned to any

segment, a new segment is provided. In all previous cases, aggregation is carried out as long as  $\delta_{i,j}$  holds, otherwise, different segments are assigned to previous pair of nodes.

In addition, the number of assigned graphs in scale  $s$  is given by  $p$ , whose description is provided in Equation 5, which depends on  $\bar{\delta}_{i,j} = 1 - \delta_{i,j}$  as follows:

$$p = p + \bar{\delta}_{i,j} + \xi_{i,j} \quad (5)$$

where function  $\xi_{i,j}$  is defined as

$$\xi_{i,j} = \begin{cases} \bar{\delta}_{i,j}, & \nexists n, m, \in \mathbb{N}, m \neq n, v_i \in \mathcal{G}_n^{[s]}, v_j \in \mathcal{G}_m^{[s]} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

This assignment is done for each value in the ordered set  $\Omega_{\mathcal{V}}$ , until whether every element in  $\Omega_{\mathcal{V}}$  is evaluated or every node in  $\mathcal{V}$  is assigned a segment in subsequent scale.

Gaussian Multiscale Aggregation assures that every node in scale  $s-1$  is assigned a segment/subgraph in scale  $s$ .

After aggregation, nodes in scale  $s$  are gathered into  $p$  subgraphs, with  $p < N^{[s]}$ , being  $N^{[s]}$  the number of nodes in scale  $s$ . Each subgraph contains a set of nodes, whose number is unknown a priori. These subgraphs must be compared in subsequent scales, and thus the similarity function in subgraphs is defined in Equation 7.

Consequently, let  $\mathcal{G}_n^{[s+1]}$  be the  $n$ th aggregated graph in scale  $s+1$ , which gathers a set of  $m$  subgraphs  $\{\mathcal{G}_1^{[s]}, \dots, \mathcal{G}_m^{[s]}\}$  in scale  $s$ . Then the similarity function for graph  $\mathcal{G}_n^{[s+1]}$ , namely  $\phi_{\mathcal{G}_n^{[s+1]}}$  is defined as

$$\phi_{\mathcal{G}_n^{[s+1]}} = \frac{\sum_{i=1}^m \phi_{\mathcal{G}_i^{[s]}}}{\int_{\alpha} \sum_{i=1}^m \phi_{\mathcal{G}_i^{[s]}} d\alpha} \quad (7)$$

Notice that the definition of the similarity functions  $\phi_{\mathcal{G}_n^{[s+1]}}$  has sense also for individual nodes in  $\mathcal{V}$ , considering nodes as graphs of one element. This is essential during the aggregation in first scale, where graphs gathers nodes instead of subgraphs. In this case, function  $\phi^{[0]}$  is represented by a gaussian function of mean and deviation corresponding to the average and dispersion intensity of their neighbour nodes, as stated before.

Therefore, similarity functions can be completely defined as in Equation 8

$$\phi_{\mathcal{G}^{[s+1]}} = \begin{cases} \mathcal{N}(\mu, \sigma) & s = 0 \\ \frac{\sum_{i=1}^m \phi_{\mathcal{G}_i^{[s]}}}{\int_{\alpha} \sum_{i=1}^m \phi_{\mathcal{G}_i^{[s]}} d\alpha} & s > 0 \end{cases} \quad (8)$$

where  $\mathcal{N}(\mu, \sigma)$  stands for the gaussian distribution given an average  $\mu$  and a  $\sigma$ , both of them corresponding to their respective neighbour properties. For clarity sake, first scale ( $s = 0$ ) is obtained based on nodes  $v \in \mathcal{V}$  and subsequent scales are obtained by gathering subgraphs.

Concerning location, the position of subgraphs is obtained by averaging the position of the nodes contained on each subgraph. This is essential in order to provide a neighbourhood structure, since after aggregation every scale  $s$  collects a scatter set of subgraphs. This structure is given by means of Delaunay triangulation.

A Delaunay graph for a set  $S = \{p_1, \dots, p_n\}$  of points in the plane has the set  $S$  as its vertices. Two vertices  $p_i$  and  $p_j$  are joined by a straight-line (representing an edge) if and only if the Voronoi regions  $V(p_i)$  and  $V(p_j)$  share an edge. In addition, for a set of points in  $\mathbb{R}^2$ , knowing the locations of the endpoints permits a solution in  $O(n \log n)$  time. Therefore, Delaunay triangulation is a suitable method to provide a neighbourhood structure to previous aggregated subgraphs.

This operation represents the final step in the loop, since at this moment, there exist a new subgraph  $\mathcal{G}^{[s+1]} = \bigcup_k \mathcal{G}_k^{[s+1]}$  at scale  $s + 1$  where each  $\mathcal{G}_k^{[s+1]}$  represents a node, and edges  $\mathcal{E}^{[s+1]}$  are provided by Delaunay triangulation, and weights  $\mathcal{W}^{[s+1]}$  are obtained based on Equations 1 and 8.

The whole loop is repeated until only two subgraphs remain, as stated at the beginning of this section. However, due to the constraints provided to aggregate (Equation 9), the method could not aggregate more segments, without achieving the goal of dividing image into two subgraphs. Therefore, Equation 3 is in practice relaxed and stated as follows in Equation 9:

$$\sigma_{i,j}^{[s+1]} \leq \sqrt{\sigma_i^{[s]} \sigma_j^{[s]}} + k^{[s]} \quad (9)$$

being  $k^{[s]}$  a factor able to avoid aggregation method from being stuck in the loop. This factor can be dynamically increased or decreased, according to previous method necessities. However, this value is initially set to  $k^{[s]} = 0.01$ , for each scale  $s$ . The capability of  $k^{[s]}$  to adapt the necessities of the algorithm remains as future work.

The computational cost of this algorithm is quasi-linear with the number of pixels, since each scale gathers nodes in the sense that nodes in subsequent scales are reduced by (in practice) a three times factor (Figure 4). Therefore, time to process the first scale (which contains the highest number of nodes) is greater than the rest of times to process subsequent scales, and the total time is approximately comparable to two times the processing time to aggregate first scale. This statement will be supported within the results Section 5.

## 4. Database

After presenting the algorithm, next section describes the creation of the database involved in evaluation.

This section describes the creation of a synthetic database containing a total of 408000 images of hands with a wide range of possible backgrounds like carpets, fabric, glass, grass, mud, different objects, paper, parquet, pavement, plastic, skin and fur, sky, soil, stones, tiles, tree, wall and wood.

The main aim of this database is twofold:

- First, the main purpose is to provide a comparative evaluation frame for segmentation algorithm, where existing approaches in literature could be compared. In other words, this database makes it possible to assess to what extent the segmentation algorithm can satisfactory perform a hand isolation from background on real scenarios.
- In addition, this database contains the ground-truth result for each image, providing a possible supervised evaluation criteria. These ground-truth images were obtained, given that hands were taken with a blue-coloured background, so that hand can be easily extracted by simple thresholding [21].



The creation of the synthetic database (named GB2S Database) considers the hands extracted in former database and the set of the aforementioned different textures, which were obtained from the website <http://mayang.com/textures/>.

First of all, a straightforward segmentation was carried out with a threshold-based segmentation [21], obtaining two binary masks:  $M_h$ , corresponding to those pixels representing hand, and  $M_b$  with pixels corresponding to background.

Afterwards, both masks are laid one over each other, with  $M_b$  containing pixels associated to a specific texture, resulting in an image with the hand over a desired background (grass, water, wood and so forth).

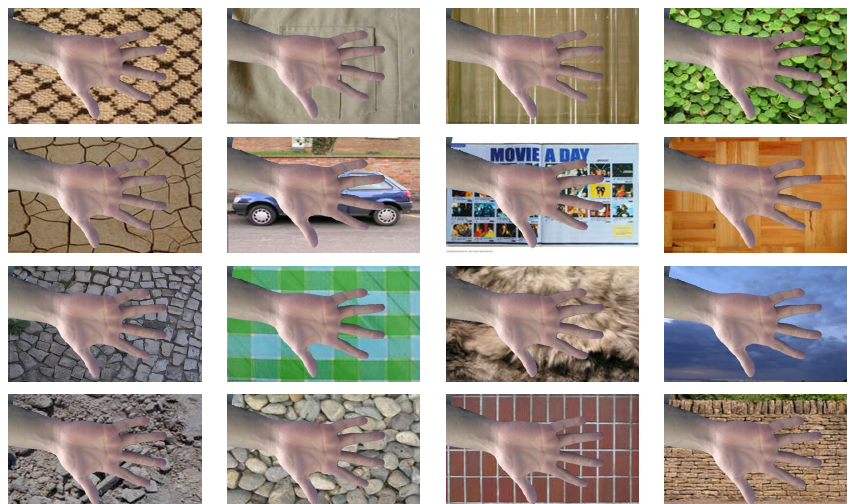
In order to ensure there is no considerable difference in illumination between hand and background, each image is converted from RGB to YCbCr color space [6] carrying out a histogram equalization in terms of illumination (Y), performing afterwards the inverse transform from YCbCr to RGB color space. Finally, a morphological operation consisting on an opening operator with a structural element of a disk of small size (5 pixels radio) is considered to fade the boundary between hand and background, so that hand is integrated within background.

All these former operations attempt to ensure a fair scenario, simulating the conditions provided in real situations.

For each hand image, a total of  $5 \times 17$  (five images and 17 textures) synthetic images are created, collecting a total of  $120 \times 2 \times 20 \times 5 \times 17 = 408000$  images (120 individuals, two hands, 20 acquisitions per hand, five images and 17 textures) to properly evaluate segmentation on real scenarios. Some visual examples of this database are provided in Figure 2.

This presented database is publicly available at <http://www.gb2s.es>.

**Figure 2.** Samples from the synthetic database in different backgrounds for a given acquisition.



Once the database has been presented, the following section comes out with the evaluation of the algorithm and the obtained results.



## 5. Results and Discussion

This section contains the results of the comparative evaluation of the proposed approach to LDC [4] and NCut [5]. First, the evaluation criteria is stated in order to provide a comparative frame, providing afterwards the results obtained in the evaluation.

### 5.1. Evaluation Criteria

Although there exist some unsupervised evaluation methods for image segmentation [34–36], we have preferred a supervised segmentation, since the synthetic database GB2S contains the corresponding ground-truth associated to each image. Segmentation results will be compared to this ground-truth image.

The proposed evaluation method is based on F-measure, [37], defined as follows:

$$F = \frac{2RP}{R + P} \quad (10)$$

where  $P$  (Precision or Confidence) stands for the number of true positives (true segmentation, i.e. classify a hand pixel as hand) in relation to the number of true positives and false negatives (false hand segmentation), and  $R$  (Recall or Sensitivity) represents the number of true positives in relation to the number of true positives and false positives (false background segmentation, i.e. consider background as hand). F-measure is within the  $[0, 1]$  interval, so that 0 states a bad segmentation, while on the contrary 1 represents the best segmentation result.

Aiming a fair comparison, the propose algorithm is compared to two competitive segmentation methods existing in the literature, namely Lossy Data Compression (LDC) [4] and Normalized Cuts (NCut) [5].

### 5.2. Gaussian Multiscale Aggregation evaluation

The evaluation of a segmentation method involves different aspects concerning accuracy, computational cost and parameters dependency.

First aspect is related to what extent the algorithm is able to properly detect or isolate a specific object within an image. Concretely in this paper, accuracy is understood as the capability of the proposed algorithm to properly isolate hand from background.

Table 1 shows the results in terms of F-measure of the proposed methods in comparison to LDC approach and Normalized Cuts. Although the results obtained by the proposed method (first column) can be improved, they overcome the other two schemes. Reader may notice that scenarios with textures similar to hand (e.g., soil) decrease the performance of the segmentation algorithms, but the proposed method still provides F-measure rates of more than 88%.

In addition, accuracy can be also visually evaluated. Figure 3 presents a comparative frame for segmentation evaluation, comparing the results obtained for the LDC method, Normalized Cuts and the proposed method. Reader can compare the obtained results (columns 4-6) to the ground-truth (column 2). The results obtained by the proposed approach conserve more precisely the shape of the hand even in scenarios with similar textures like parquet (row 5) or wood (last row).

**Table 1.** Segmentation evaluation by means of F-measure in database GB2S with 17 different background textures, together with the corresponding standard deviation. In addition, the results for LDC and NCut are also provided for comparison.

Texture	Proposed, $F(\%)$	LDC, $F(\%)$	NC, $F(\%)$
Carpets	92.1±0.1	73.7±0.3	65.1±0.3
Paper	91.3±0.1	83.2±0.2	72.8±0.4
Stones	91.2±0.1	78.2±0.4	71.5±0.3
Fabric	88.4±0.3	65.3±0.1	60.1±0.2
Parquet	88.3±0.2	66.1±0.2	62.3±0.3
Tiles	90.1±0.2	71.5±0.3	68.7±0.2
Glass	94.1±0.1	75.8±0.1	71.4±0.1
Pavement	88.9±0.2	67.8±0.1	63.7±0.2
Tree	96.0±0.2	73.4±0.2	67.2±0.1
Grass	93.3±0.2	70.1±0.1	65.3±0.2
Skin and Fur	95.3±0.3	82.3±0.2	71.8±0.3
Wall	94.1±0.1	70.9±0.2	62.3±0.2
Mud	89.5±0.2	68.3±0.1	60.1±0.2
Sky	96.1±0.1	77.2±0.2	71.3±0.1
Wood	93.5±0.1	82.5±0.2	73.5±0.1
Objects	92.0±0.1	70.1±0.1	61.6±0.3
Soil	89.0±0.2	67.2±0.3	59.7±0.2

**Table 2.** Relation between time performance (in seconds), the dimension of the image, and the size in number of pixels, comparing the proposed method with LDC approach and Normalized Cuts (NCut).

Image Dimensions	Number of Pixels	Proposed (seconds)	LDC (seconds)	NCut (seconds)
600x800	480000	30.1	233.1	321.7
450x600	270000	19.8	63.4	129.5
300x400	120000	9.4	52.1	25.1
150x200	30000	3.1	32.8	7.2

Secondly, concerning computational cost, Table 2 presents the segmentation time in relation to the number of pixels of the images. This temporal evaluation was carried out in a PC computer @2.4 GHz Intel Core 2 Duo with 4GB 1067 MHz DDR3 of memory, considering that the proposed method was completely implemented in MATLAB.

The results provided in Table 2 shows that the proposed algorithm is faster than the compared approaches. In addition, the proposed method can segment images of higher sizes, but LDC and NCut cannot handle higher sizes images without running out of memory.

Finally, this section will study the dependency of two parameters strongly related to algorithm performance, namely  $k$  factor and aggregation linearity (Equations 10 and 9).

Factor  $k$  controls the aggregation capability of the overall method. Within these experiments, factor  $k$  was experimentally fixed to  $k = 0.01$ , ensuring that the number of segments in the last scale is two: hand and background. However, extending the proposed approach to other applications in image processing would imply to provide a dynamic factor  $k$ , depending whether the algorithm standstills in a certain scale. The proposal of a dynamic factor  $k$  remains as future work.

Figure 4 presents the relation between number of segments along scales using different values of  $k$ . Notice that  $k = 0$  implies no stopping criteria, and therefore aggregates scales until only one segment is obtained.

During the explanation of the method, the algorithm is said to be quasi-linear with the number of pixels. This statement is supported by Table 2, but for clarity sake, we provide a chart (Figure 5) indicating which proportion of time is required for each scale. The most demanding scale is the first one, whose proportion is higher than the other parts, concluding that the algorithm has indeed a quasi-linear behaviour in relation to the number of pixels.

## 6. Conclusions and Future Work

The application of hand biometrics to unconstrained and contact-less, platform-free environments implies an increase in difficulty in the pre-processing and segmentation procedure in hand acquisition. Therefore, an unsupervised segmentation algorithm has been proposed based on gaussian multiscale aggregation. This method gathers iteratively those pixels similar in texture and color under segments, until a certain number of clusters/segments is provided as a result.

This method is able to isolate hand from a wide range of backgrounds (carpets, fabric, glass, grass, mud, different objects, paper, parquet, pavement, plastic, skin and fur, sky, soil, stones, tiles, tree, wall and wood), simulating real situations and unconstrained background scenarios.

Besides, the evaluation of the proposed approach has been carried out based on a publicly available synthetic database, containing 408000 hand image acquisitions with different background textures. The evaluation consisted of a comparison of the performance in terms of accuracy and computational cost to two competitive segmentation methods existing in literature, namely Lossy Data Compression (LDC) [4] and Normalized Cuts (NCuts) [5].

The results obtained point out that the performance of the proposed algorithm outcomes existing segmentation algorithms in literature, regarding not only accuracy and computational cost, but also memory usage, since the proposed algorithm is quasi-linear in relation to the number of pixels.

As future work, we consider to implement the method with a dynamic  $k$  parameter, so that the algorithm can be adapted to any image, providing segmentations of more complex images. In addition, we aim a faster implementation of the method considering both software and hardware optimized implementation, together with a more complete evaluation with other publicly available databases.

## Acknowledgements

This research has been supported by the Ministry of Industry, Tourism and Trade of Spain, in the framework of the project CENIT-Segur@, reference CENIT-2007 2004.

## References

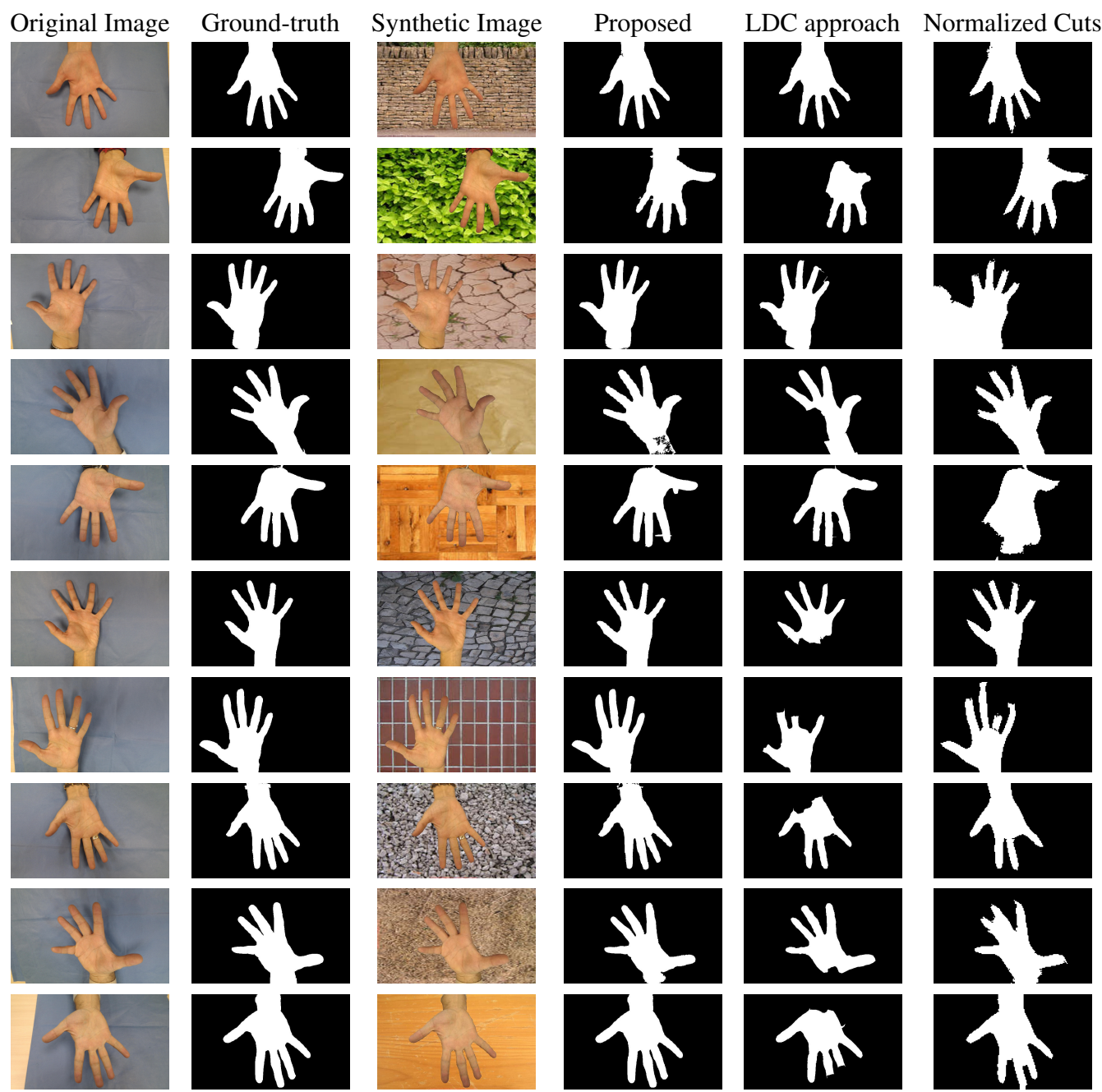
1. Kukula, E.; Elliott, S. Implementation of hand geometry at Purdue University's Recreational Center: an analysis of user perspectives and system performance. *Security Technology*, 2005. CCST '05. 39th Annual 2005 International Carnahan Conference on, 2005, pp. 83–88.
2. Kukula, E.; Elliott, S. Implementation of hand geometry: an analysis of user perspectives and system performance. *Aerospace and Electronic Systems Magazine, IEEE* **2006**, *21*, 3–9.
3. Muñoz, A.G.C.; Ávila, C.S.; de Santos Sierra, A.; Casanova, J.G. A mobile-oriented hand segmentation algorithm based on fuzzy multiscale aggregation. *Proceedings of the 6th international conference on Advances in visual computing - Volume Part I*; Springer-Verlag: Berlin, Heidelberg, 2010; ISVC'10, pp. 479–488.
4. Ma, Y.; Derksen, H.; Hong, W.; Wright, J. Segmentation of Multivariate Mixed Data via Lossy Data Coding and Compression. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2007**, *29*, 1546–1562.
5. Shi, J.; Malik, J. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2000**, *22*, 888–905.
6. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing (3rd Edition)*; Prentice-Hall, Inc.: Upper Saddle River, NJ, USA, 2006.
7. Morales, A.; Ferrer, M.; Alonso, J.; Travieso, C. Comparing infrared and visible illumination for contactless hand based biometric scheme. *Security Technology*, 2008. ICCST 2008. 42nd Annual IEEE International Carnahan Conference on, 2008, pp. 191–197.

8. Hennings-Yeomans, P.; Kumar, B.; Savvides, M. Palmprint Classification Using Multiple Advanced Correlation Filters and Palm-Specific Segmentation. *Information Forensics and Security, IEEE Transactions on* **2007**, *2*, 613–622.
9. Wang, Y.; Wang, H. Gradient Based Image Segmentation for Vein Pattern. *Computer Sciences and Convergence Information Technology*, 2009. ICCIT '09. Fourth International Conference on, 2009, pp. 1614–1618.
10. Segundo, M.; Silva, L.; Bellon, O.; Queirolo, C. Automatic Face Segmentation and Facial Landmark Detection in Range Images. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* **2010**, *40*, 1319–1330.
11. He, Z.; Tan, T.; Sun, Z.; Qiu, X. Toward Accurate and Fast Iris Segmentation for Iris Biometrics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2009**, *31*, 1670–1684.
12. Yan, P.; Bowyer, K. Biometric Recognition Using 3D Ear Shape. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2007**, *29*, 1297–1308.
13. Huang, L.; Xu, Z.; Hu, F. A novel gait contours segmentation algorithm. *Computer, Mechatronics, Control and Electronic Engineering (CMCE)*, 2010 International Conference on, 2010, Vol. 6, pp. 410–413.
14. Graves, A.; Liwicki, M.; Fernandez, S.; Bertolami, R.; Bunke, H.; Schmidhuber, J. A Novel Connectionist System for Unconstrained Handwriting Recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2009**, *31*, 855–868.
15. Sanchez-Reillo, R.; Sanchez-Avila, C.; Gonzalez-Marcos, A. Biometric identification through hand geometry measurements. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2000**, *22*, 1168–1171.
16. Lew, Y.; Ramli, A.; Koay, S.; Ali, R.; Prakash, V. A hand segmentation scheme using clustering technique in homogeneous background. *Research and Development*, 2002. SCORed 2002. Student Conference on, 2002, pp. 305–308.
17. Woodard, D.L.; Flynn, P.J. Finger surface as a biometric identifier. *Comput. Vis. Image Underst.* **2005**, *100*, 357–384.
18. Zheng, G.; Wang, C.J.; Boulton, T. Application of Projective Invariants in Hand Geometry Biometrics. *Information Forensics and Security, IEEE Transactions on* **2007**, *2*, 758–768.
19. Morales, A.; Ferrer, M.A.; Daz, F.; Alonso, J.B.; Travieso, C.M. Contact-Free Hand Biometric System for Real Environments. *European Conference on Signal Processing 2008* **2008**.
20. Kumar, A.; Zhang, D. Personal recognition using hand shape and texture. *Image Processing, IEEE Transactions on* **2006**, *15*, 2454–2461.
21. Amayeh, G.; Bebis, G.; Erol, A.; Nicolescu, M. Peg-Free Hand Shape Verification Using High Order Zernike Moments. *Computer Vision and Pattern Recognition Workshop*, 2006. CVPRW '06. Conference on, 2006, p. 40.
22. Kumar, A.; Wong, D.C.M.; Shen, H.C.; Jain, A.K. Personal Verification using Palmprint and Hand Geometry Biometric. 2003, pp. 668–678.
23. Doublet, J.; Lepetit, O.; Revenu, M. Contactless Hand Recognition Based on Distribution Estimation. *Biometrics Symposium*, 2007, 2007, pp. 1–6.

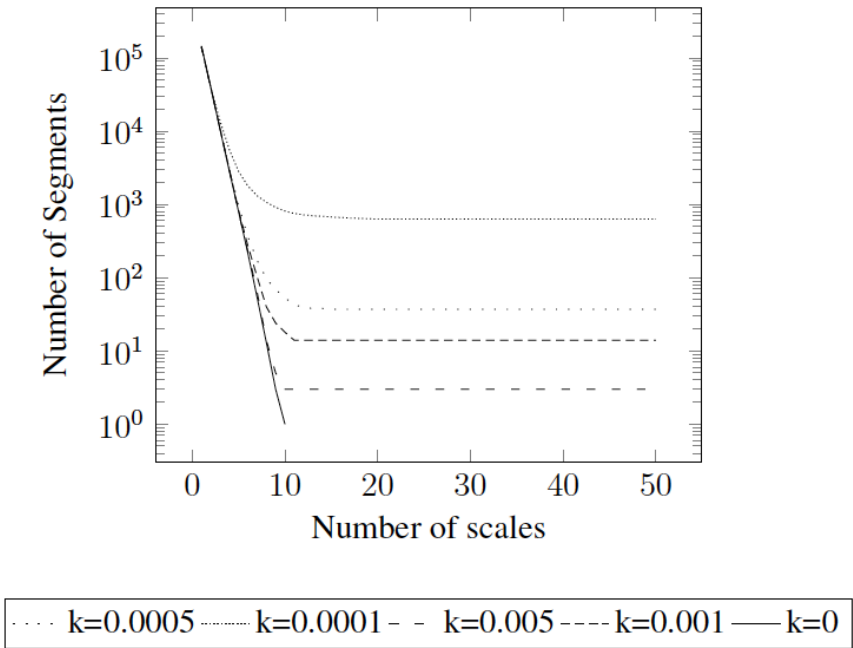
24. de Santos Sierra, A.; Casanova, J.; Avila, C.; Vera, V. Silhouette-based hand recognition on mobile devices. *Security Technology*, 2009. 43rd Annual 2009 International Carnahan Conference on, 2009, pp. 160–166.
25. Jain, A.; Ross, A.; Pankanti, S. A prototype hand geometry-based verification system. *Proceedings of Second International Conference on Audio- and Video-based Biometric Person Authentication*, 1999, Vol. 1, pp. 166–171.
26. Galun, M.; Apartsin, A.; Basri, R. Multiscale segmentation by combining motion and intensity cues. *Computer Vision and Pattern Recognition*, 2005. CVPR 2005. IEEE Computer Society Conference on, 2005, Vol. 1, pp. 256–263 vol. 1.
27. Sharon, E.; et al.. Fast Multiscale Image Segmentation.
28. Gauch, J. Image segmentation and analysis via multiscale gradient watershed hierarchies. *Image Processing, IEEE Transactions on* **1999**, 8, 69–79.
29. Vanhamel, I.; Pratikakis, I.; Sahli, H. Multiscale gradient watersheds of color images. *IEEE Transactions on Image Processing* **2003**, 12, 617–626.
30. Felzenszwalb, P.F.; Huttenlocher, D.P. Efficient graph-based image segmentation. *International Journal of Computer Vision* **2004**, 59, 2004.
31. Galun, M.; Sharon, E.; Basri, R.; Brandt, A. Texture segmentation by multiscale aggregation of filter responses and shape elements. *IN ICCV*, 2003, pp. 716–723.
32. Chen, S.; Cao, L.; Wang, Y.; Liu, J.; Tang, X. Image segmentation by MAP-ML estimations. *Trans. Img. Proc.* **2010**, 19, 2254–2264.
33. Berg, M.; Cheong, O.; Kreveld, M. *Computational geometry: algorithms and applications*; Springer, 2008.
34. Chen, S.; Cao, L.; Wang, Y.; Liu, J.; Tang, X. Image Segmentation by MAP-ML Estimations. *Image Processing, IEEE Transactions on* **2010**, 19, 2254–2264.
35. Unnikrishnan, R.; Pantofaru, C.; Hebert, M. Toward Objective Evaluation of Image Segmentation Algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, 29, 929–944.
36. Meilă, M. Comparing clusterings: an axiomatic view. *Proceedings of the 22nd international conference on Machine learning*; ACM: New York, NY, USA, 2005; ICML '05, pp. 577–584.
37. Alpert, S.; Galun, M.; Basri, R.; Brandt, A. Image Segmentation by Probabilistic Bottom-Up Aggregation and Cue Integration. *IEEE Conference on Computer Vision and Pattern Recognition*, 2007. CVPR '07., 2007, pp. 1–8.



**Figure 3.** A comparative study of results provided by segmentation algorithm in comparison to ground-truth. First column gathers examples from first database, together with their segmentation on second column, considered as ground truth. Third column presents synthetic images based on first column images, providing on the fourth column the final segmentation result. Last two column present the segmentation result provided by the Lossy Data Compression (LDC) [4] and Normalized Cuts [5], respectively.



**Figure 4.** Dependency of the aggregation process on parameter  $k$ . The lower  $k$ , the lower the constraints to aggregate segments. Notice that  $k = 0$  means no stopping condition.



**Figure 5.** Proportion of processing time for each scale. Most of the time is required by the aggregation procedure on the first scale.

